

1 CLAIMS

2 1. An integrated circuit (IC) card comprising:

3 an input/output (I/O) interface; and

4 a smart card development interface, coupled to the I/O interface, to receive
5 and identify one or more debug frames interlaced within a normal communication
6 flow between the IC card and a host system.

7
8 2. An IC card according to claim 1, further comprising:

9 a memory device having stored therein a plurality of executable
10 instructions; and

11 a controller, coupled to the memory device and the smart card development
12 interface, to execute at least a subset of the plurality of executable instruction to
13 selectively implement one or more of a plurality of IC card applets.

14
15 3. An IC card according to claim 2, wherein the memory device
16 includes a plurality of executable instructions which, when executed, implement a
17 debug application which selectively controls other applications executing on the
18 IC card.

19
20 4. An IC card according to claim 1, wherein the smart card development
21 interface includes a debug filter to identify and remove the debug frames from the
22 normal communication flow.
23
24
25

1 5. An IC card according to claim 4, wherein the debug filter redirects
2 the debug frames to a debug application on the IC card.

3
4 6. An IC card according to claim 1, further comprising a debug
5 application, responsive to debug instructions embedded within received debug
6 frames, the debug application providing a user with a host of application debug
7 features enabled in response to the received debug instructions.

8
9 7. An IC card according to claim 6, wherein select debug instructions
10 invoke one or more of the following debug features: read/write IC card memory,
11 get/set breakpoints in an IC card applet, sequentially step an IC card application,
12 run an IC card applet, and release an IC card applet frame.

13
14 8. An IC card according to claim 1, wherein the IC card communicates
15 with a remote host system using a transport protocol comprising application data
16 units (APDU) and debug protocol data units (DPDU).

17
18
19 9. An IC card according to claim 8, wherein the transport protocol is a
20 standard smart card communication protocol, wherein the APDU and the DPDU
21 adhere to the standard IC card communication protocol.

1 10. An IC card according to claim 8, wherein the smart card
2 development interface further comprises a debug filter which identifies DPDU
3 within the normal communication flow to redirect the DPDU to a debug
4 application on the IC card.

5
6 11. An IC card according to claim 10, wherein the debug filter identifies
7 DPDU within the normal communication flow by detecting an invalid source
8 and/or destination address identifier within the debug frame.

9
10 12. An IC card according to claim 1, wherein the IC card communicates
11 with a remote host system using a transport protocol comprising application data
12 units (APDU) including normal application frames and debug frames.

13
14 13. A storage medium having stored thereon a plurality of executable
15 instructions which, when executed, implement the smart card development
16 interface of claim 1.

17
18 14. A method of debugging a smart card application, the method
19 comprising:

20 receiving one or more debug frames interlaced with application frames
21 comprising a normal communication flow between a smart card and a host system;

22 identifying the one or more debug frames;

23 routing the received debug frames to a debug application executing on the
24 smart card, while promoting the application frames to an application executing on
25 the smart card, subject to conditions imposed by the debug frames.

1
2 15. A method according to claim 14, wherein the step of identifying the
3 one or more debug frames comprises:

4 reading a source and/or destination address of frames comprising the
5 normal communication flow; and

6 detecting invalid source and/or destination addresses in select frames
7 denoting debug frames.

8
9 16. A method according to claim 14, further comprising:
10 implementing one or more debug features on the smart card according to
11 debug instructions embedded within the received debug frames.

12
13 17. A method according to claim 16, wherein the debug features include
14 one or more of read/write smart card memory, get/set breakpoints in a smart card
15 application, sequentially step a smart card application, run a smart card
16 application, and release a smart card application frame.

17
18 18. A method according to claim 14, further comprising:
19 generating a response debug frame to a received debug frame;
20 interlacing the response debug frame with response application frames; and
21 sending the response debug frame and response application frames to a host
22 system.

1 19. A storage medium having stored thereon a plurality of instructions
2 which, when executed, implement the method of claim 14.

3
4 20. A computer system comprising:
5 an input/output (I/O) interface; and
6 a client development interface, coupled to the I/O interface, to receive and
7 identify debug frames interlaced within the normal communication flow between
8 the computer system and a removably coupled smart card.

9
10 21. A computer system according to claim 20, further comprising:
11 a memory device having stored therein a plurality of instructions; and
12 a processor, coupled to the memory device and the client development
13 interface, to execute at least a subset of the plurality of instructions to implement
14 one or more applications including a smart card development application having a
15 debug environment to send and receive debug frames to the coupled IC card
16 interlaced within the normal communication flow between the computer system
17 and the IC card.

18
19 22. A computer system according to claim 14, wherein the memory
20 device includes a plurality of executable instructions which, when executed,
21 implement a debug application on the computer system to communicate with and
22 control smart card resources.

1 23. A computer system according to claim 20, wherein the client
2 development interface includes a debug filter to identify and remove the debug
3 frames from the normal communication flow between the computer system and
4 the smart card.

5
6 24. A computer system according to claim 23, wherein the debug filter
7 redirects debug frames received from the smart card to a debug application
8 executing on the computer system.

9
10 25. A computer system according to claim 20, further comprising a
11 debug application, to write and read debug frames to and from the smart card,
12 facilitating a number of application debugging features.

13
14 26. A computer system according to claim 25, wherein the debug
15 frames written by the debug application invoke one or more of the following
16 debug features: read/write smart card memory, get/set breakpoints in a smart card
17 application, sequentially step a smart card application, run a smart card
18 application, and release a smart card application frame.

19
20 27. A computer system according to claim 20, wherein the computer
21 system communicates with the smart card using a transport protocol comprising
22 application data units (APDU) and debug protocol data units (DPDU).
23
24
25

1 28. A computer system according to claim 27, wherein the client
2 development interface further comprises a debug filter which identifies DPDU
3 within the normal communication flow to redirect the DPDU to a debug
4 application executing on the computer system.

5
6 29. A computer system according to claim 29, wherein the debug filter
7 identifies DPDU within the normal communication flow by detecting an invalid
8 source and/or destination address identifier within the debug frame.

9
10 30. A storage medium having stored thereon a plurality of executable
11 instructions which, when executed, implement the client development interface of
12 claim 20.

13
14 31. A computer-implemented method for debugging a smart card
15 application, the method comprising:

16 generating one or more debug frames containing debug instructions;

17 interlacing the generated debug frames with one or more application frames
18 generated according to an application executing on the computer; and

19 sending the application frames with the interlaced debug frames to a
20 removably coupled smart card, wherein the debug frames invoke one or more
21 debug features of the smart card.

1 32. A computer-implemented method according to claim 31, wherein
2 the application frames are generated by an application executing within an
3 application development environment, while the debug frames are generated in
4 response to user interaction with the smart card application development
5 environment.

6
7 33. A computer-implemented method according to claim 31, wherein
8 generating one or more debug frames comprises populating a source and/or
9 destination field of the debug frame with an invalid source and/or destination
10 address.

11
12 34. A computer-implemented method according to claim 31, further
13 comprising:

14 receiving a normal communication flow from the smart card including
15 debug frames interlaced with application frames, wherein the debug frames
16 received from the smart card are received in response to debug frames issued by
17 the computer.

18
19 35. A computer-implemented method according to claim 34, wherein
20 the application frames are promoted to an associated application executing within
21 an application development tool executing on the computer, while the debug
22 frames are promoted to an application debug environment of the application
23 development tool executing on the computer.
24
25

1 36. A communication protocol, employed between a host system and a
2 smart card, the protocol comprising:

3 a plurality of application frames comprising a normal communication flow
4 between a host application and a smart card application; and

5 one or more debug frames, interlaced with the application frames within the
6 normal communication flow, to enable a debug application executing on the host
7 system to selectively access and control smart card resources.

8
9 37. A communication protocol according to claim 36, wherein the
10 debug application and the host application are executing on separate host systems,
11 each communicatively coupled to the smart card.

12
13 38. A communication protocol according to claim 36, wherein the one
14 or more debug frames include debug instruction to implement one or more of the
15 following debug features: read/write smart card memory, get/set breakpoints in a
16 smart card application, sequentially step a smart card application, run a smart card
17 application, and release a smart card application frame.

18
19 39. A communication protocol according to claim 36, wherein the
20 debug frame is distinguished from an application frame by incorporating an
21 invalid source address.

1 40. A communication protocol according to claim 36, wherein the
2 debug frame is distinguished from an application frame by incorporating an
3 invalid destination address.

4
5 41. An application development system comprising:
6 a computer system to execute an application within an application
7 development tool; and
8 a smart card incorporating a smart card development interface, coupled to
9 the computer system, to receive and identify debug frames interlaced with
10 application frames within a normal communication flow between the application
11 executing on the computer system and the smart card, wherein the smart card
12 development interface promotes the application frames to an application layer of
13 the smart card, and invokes debug features of the smart card in response to debug
14 instructions embedded within the received debug frames.

15
16 42. An application development system according to claim 41, wherein
17 the computer system further comprises:

18 a client development interface, to interlace debug frames generated by the
19 application development tool with application frames generated by the application
20 executing within the application development tool.

21
22 43. An application development system according to claim 42, wherein
23 the application development tool generates debug frames in response to user
24 interaction with the application development tool.

1 44. An application development system according to claim 43, wherein
2 the application development tool populates a source and/or destination field of the
3 debug frame with an invalid source and/or destination address.

4
5 45. An application development system according to claim 43, wherein
6 the debug frames invoke and control one or more smart card resources facilitating
7 debugging of the application executing within the application development tool of
8 the computer system.

9
10 46. An application development system according to claim 42, wherein
11 the client development interface includes a debug filter to identify and route debug
12 frames received from the smart card.

13
14 47. An application development system according to claim 41, wherein
15 the smart card development interface comprises a debug filter to identify debug
16 frames within the received normal communication flow.

17
18 48. An application development system according to claim 47, wherein
19 the debug filter identifies debug frames by an invalid source and/or destination
20 address embedded within a source and/or destination field of the debug frame.

21
22 49. An application development system according to claim 41, further
23 comprising:

24 a communication protocol, employed by the computer system and the smart
25 card to communicate therebetween, the communication protocol comprising,

~~one or more debug frames, interlaced with the application frames within the normal communication flow, to enable a debug application executing on the host system to selectively access and control smart card resources.~~

~~one or more debug frames, interlaced with the application frames within the normal communication flow, to enable a debug application executing on the host system to selectively access and control smart card resources.~~